


bpmonline



Разработка приложений для marketplace



Будущее создавать просто

Содержание

1.	Разработка приложений для marketplace	2
1.1.	Настройка профиля разработчика	2-3
1.2.	Заказ сайта для разработки	3-5
1.3.	Пример разработки простого приложения для bpm'online marketplace	5-13
1.4.	Перенос решения на тестовый сайт	13-15
1.5.	Регистрация решения в Личном кабинете	15-22
1.6.	Публикация решения в marketplace	22-23
1.7.	Примеры разработки приложений	23
1.7.1.	Пример разработки приложения с пользовательским разделом	23-31
1.7.2.	Привязка данных к пакету	31-36

1 Разработка приложений для marketplace

Общие сведения

Bpm'online marketplace — это онлайн-площадка, где любой пользователь может легко найти и заказать готовое решение для своего бизнеса. marketplace является точкой контакта между клиентом и разработчиком с целью ознакомления, выбора и приобретения партнерских решений.

В marketplace могут быть опубликованы два типа партнерских решений:

- Вертикальное решение — партнерская конфигурация, разработанная на базе продуктов bpm'online, закрывающая потребность конкретной индустрии и имеющая самостоятельную бизнес-ценность.
- Приложение — решение, расширяющее возможности базовых продуктов bpm'online или вертикальных решений и имеющее дополнительную бизнес-ценность.

Подробнее о bpm'online marketplace и его элементах можно узнать из регламента выпуска партнерских решений, доступного на [странице регистрации разработчика marketplace](#).

Сразу же после регистрации нового разработчика для него становится доступен Личный кабинет разработчика. Все действия, необходимые для регистрации и публикации приложения в bpm'online marketplace выполняются только в Личном кабинете.

Последовательность регистрации и публикации пользовательского приложения

1. Настроить профиль разработчика. Настройки профиля разработчика содержат информацию, которая будет отображена в витрине marketplace для каждого решения. О том, как выполнить настройки профиля разработчика, подробно изложено в статье "**Настройка профиля разработчика (Section 1.1)**".
2. Заказать сайт для разработки. Разработку пользовательского решения необходимо вести в среде разработки, которая представляет собой отдельный сайт. Действия, необходимые для заказа такого сайта, описаны в статье "**Заказ сайта для разработки (Section 1.2)**".
3. Выполнить разработку приложения. На этом шаге выполняется разработка функциональности пользовательского решения. Подробный пример интеграции стороннего пользовательского сервиса в решение для bpm'online marketplace описан в статье "**Пример разработки простого приложения для bpm'online marketplace (Section 1.3)**". Более сложные примеры приведены в разделе "**Примеры разработки приложений (Section 1.7)**".
4. Выполнить тестирование. Прежде чем публиковать разработанное приложение в bpm'online marketplace, необходимо удостовериться в его работоспособности на тестовом сайте. Процесс переноса пакета с разработанным решением подробно описан в статье "**Перенос решения на тестовый сайт (Section 1.4)**".
5. Зарегистрировать разработанное решение в Личном кабинете. Для предварительной проверки разрабатываемого решения службой поддержки marketplace, предложение необходимо зарегистрировать в Личном кабинете. Как это сделать, подробно описывается в статье "**Регистрация решения в Личном кабинете (Section 1.5)**".
6. Опубликовать приложение. Особенности публикации приложения в bpm'online marketplace подробно изложены в статье "**Публикация решения в marketplace (Section 1.6)**".

1.1 Настройка профиля разработчика

Настройки профиля разработчика должны содержать актуальную информацию, которая будет отображена в витрине marketplace для каждого решения. Эти данные необходимо внести еще до регистрации первого решения в Личном кабинете.

Для того чтобы заполнить данные о разработчике решения, необходимо в Личном кабинете перейти в раздел [Настройки профиля] > [Профиль разработчика] (рис. 1).

Рис. 1. — Раздел [Профиль разработчика]

Приложения

Быстрый старт

Настройки профиля


Профиль разработчика

Профиль разработчика
Учасники
Настройки пользователя

Название *

Terrasoft Academy

Логотип



×

Префикс разработчика

tsa

В разделе [Профиль разработчика] необходимо заполнить следующие поля:

- Название — название организации, от имени которой разработанные решения будут публиковаться на marketplace. Если разработка и публикация решений выполняются частным лицом, в названии могут быть указаны фамилия, имя и отчество разработчика. Это поле обязательно для заполнения.
- Логотип — корпоративный логотип разработчика. Рекомендуется использовать изображение на белом фоне в формате .png, .gif, .jpg, .jpeg с разрешением 200 px по ширине.
- Префикс разработчика — это уникальный префикс, который устанавливается в названиях пользовательских схем, пакетов, а также в названиях пользовательских колонок в объектах-наследниках системных объектов и позволяет определить функциональность, созданную разработчиком. Префикс должен содержать цифры или буквы латинского алфавита. Длина префикса — от трех до пятнадцати символов. После регистрации префикс не может быть изменен.
- Описание — краткая информация, описывающая направления деятельности и компетенции разработчика.
- Ссылка на вебсайт — ссылка на Web-сайт разработчика.
- Контакты разработчика — группа полей, в которых необходимо указать контактные данные разработчика, необходимые для коммуникации клиента с разработчиком и предоставления поддержки пользователям партнерского решения.
- Контакты поддержки — группа полей, в которых указываются средства связи службы поддержки организации-разработчика, если такая служба существует.

1.2 Заказ сайта для разработки

Чтобы иметь возможность разрабатывать собственные приложения на платформе bpm'online, необходимо предварительно развернуть облачную среду разработки — отдельное приложение bpm'online, которое предназначено для создания новой функциональности разработчиками. Подробнее рабочие среды и организация процесса разработки bpm'online рассмотрены в статье "[Организация процесса разработки](#)".

Облачная среда разработки является набором (bundle) из трех флагманских продуктов bpm'online sales,

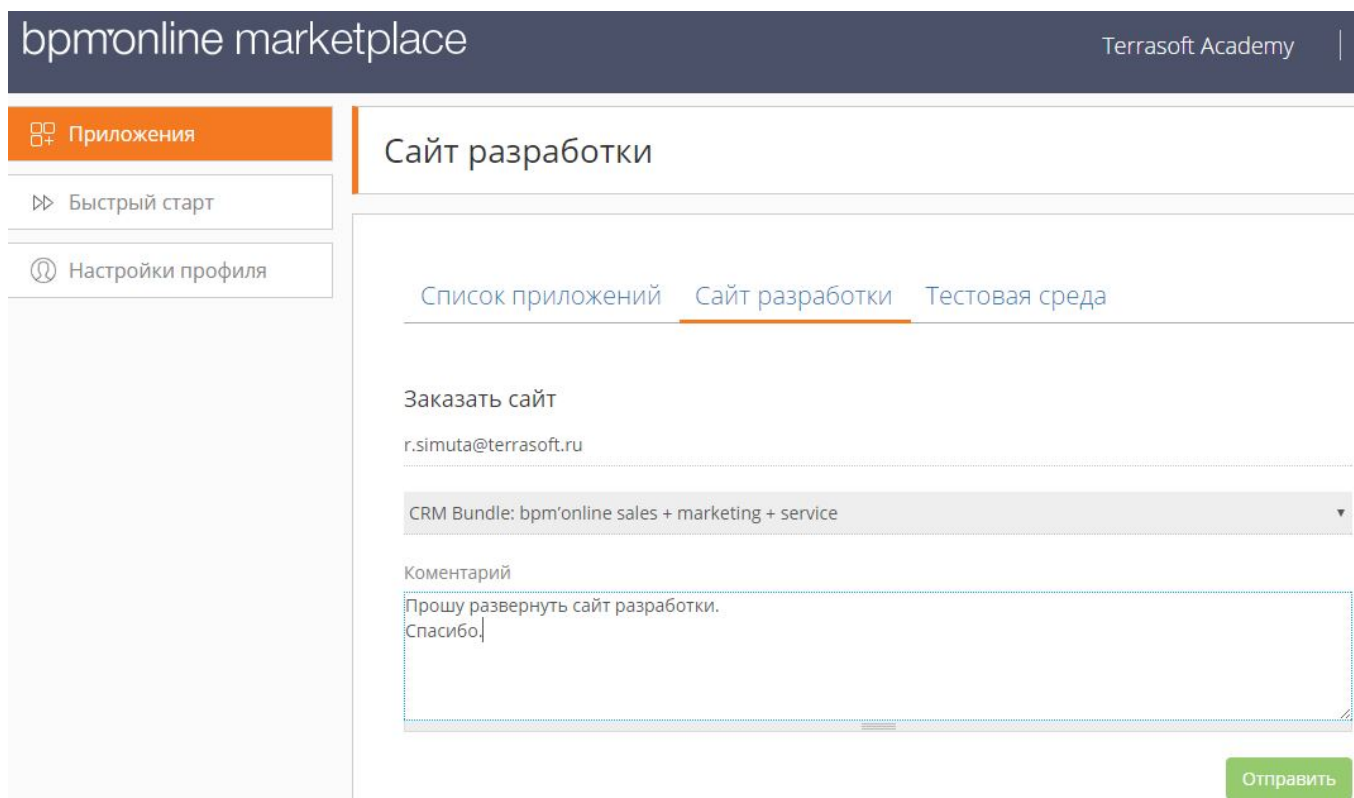
bpm'online marketing и bpm'online service. Это позволяет использовать любую базовую функциональность bpm'online при разработке собственных решений.

ВАЖНО

Перед заказом сайта для разработки нужно обязательно установить [Префикс разработчика] в разделе [Настройки профиля] > [Профиль разработчика]. Подробности см. "**Настройка профиля разработчика (Section 1.1)**".

Для развертывания нового сайта разработки необходимо использовать форму заказа сайта в Личном кабинете. Для этого следует перейти в раздел [Приложения] и на вкладке [Сайт разработки] выполнить запрос в службу поддержки (рис. 1).

Рис. 1. — Форма заказа сайта разработки



The screenshot displays the 'bpm'online marketplace' interface. At the top, the header includes the logo and 'Terrasoft Academy'. A left sidebar contains navigation options: 'Приложения' (Applications), 'Быстрый старт' (Quick Start), and 'Настройки профиля' (Profile Settings). The main content area is titled 'Сайт разработки' (Website Development) and features three tabs: 'Список приложений' (List of Applications), 'Сайт разработки' (Website Development), and 'Тестовая среда' (Test Environment). The 'Сайт разработки' tab is selected, showing a form to 'Заказать сайт' (Order website). The form includes an email field with 'r.simuta@terrasoft.ru', a dropdown menu for 'CRM Bundle' set to 'bpm'online sales + marketing + service', and a text area for 'Комментарий' (Comment) containing 'Прошу развернуть сайт разработки. Спасибо.' (Please deploy the development website. Thank you.). A green 'Отправить' (Send) button is located at the bottom right of the form.

Развертывание сайта разработки будет осуществлено службой поддержки в течение 8-ми рабочих часов с момента отправки заявки.

Когда сайт разработки будет развернут, пользователю будет отправлено автоматическое уведомление на Email-адрес, указанный в форме заказа. Ссылка на сайт разработки будет размещена и доступна в той же вкладке, из которой был осуществлен заказ сайта (рис. 2).

Рис. 2. — Ссылка на сайт разработки

The screenshot shows the bpm'online marketplace interface. At the top left is the logo 'bpm'online marketplace' and at the top right is 'Terrasoft Academy'. A left sidebar contains navigation items: 'Приложения' (Applications), 'Быстрый старт' (Quick Start), and 'Настройки профиля' (Profile Settings). The main content area is titled 'Сайт разработки' (Development Site) and includes a sub-menu with 'Список приложений' (List of Applications), 'Сайт разработки' (Development Site), and 'Тестовая среда' (Test Environment). Below this, it displays the URL for the development site: 'Сайт разработки: https://mkpdev-academy.bpmonline.com'.

1.3 Пример разработки простого приложения для bpm'online marketplace

Общие сведения

Приложение для bpm'online marketplace является проектным решением, расширяющим возможности базовых продуктов bpm'online и имеющее дополнительную бизнес-ценность. Поэтому разработка приложения для marketplace практически ничем не отличается от разработки обычного проектного решения. Подробности об основных принципах разработки решений для базовых продуктов bpm'online можно узнать в разделе "[С чего начать разработку](#)" документации по разработке.

Поскольку приложение для bpm'online marketplace является проектным решением, предоставляемым сторонним пользователям, то необходим механизм установки этого решения в пользовательские приложения. В bpm'online этот механизм реализован с помощью пакетов. Пакет bpm'online — это совокупность конфигурационных элементов, которые реализуют определенный блок функциональности. Подробнее о пакетах и работе с ними можно узнать из раздела "[Работа с пакетами](#)" документации по разработке.

Количество и состав пакетов, реализующих приложение для bpm'online marketplace зависит от сложности функциональности, заложенной в приложение. Например, для вызова стороннего сервиса с существующей страницы редактирования достаточно заместить одну или несколько схем, сгруппировав их в одном пользовательском пакете. А для создания нового раздела bpm'online, реализующего сложную функциональность, необходимо разработать уже несколько десятков новых схем и модулей.

Последовательность создания простого приложения:

1. Если необходимо, создать Web-сервис. Web-сервис может быть реализован с [помощью конфигурационных элементов bpm'online](#) или как стороннее решение.
2. Создать пользовательский пакет.
3. В созданный пакет добавить замещающую схему модели представления главного меню.
4. В замещающей схеме реализовать необходимую функциональность.

Описание приложения

Создать проектное решение, реализующее на странице главного меню приложения bpm'online для продукта sales enterprise вызов стороннего Web-сервиса, предоставляющего строку приветствия. Вызов Web-сервиса реализовать при помощи HTTP-запроса методом POST, передав в качестве параметра имя текущего пользователя. Результат отобразить под секцией ссылок на страницы социальных сетей.

Алгоритм реализации приложения

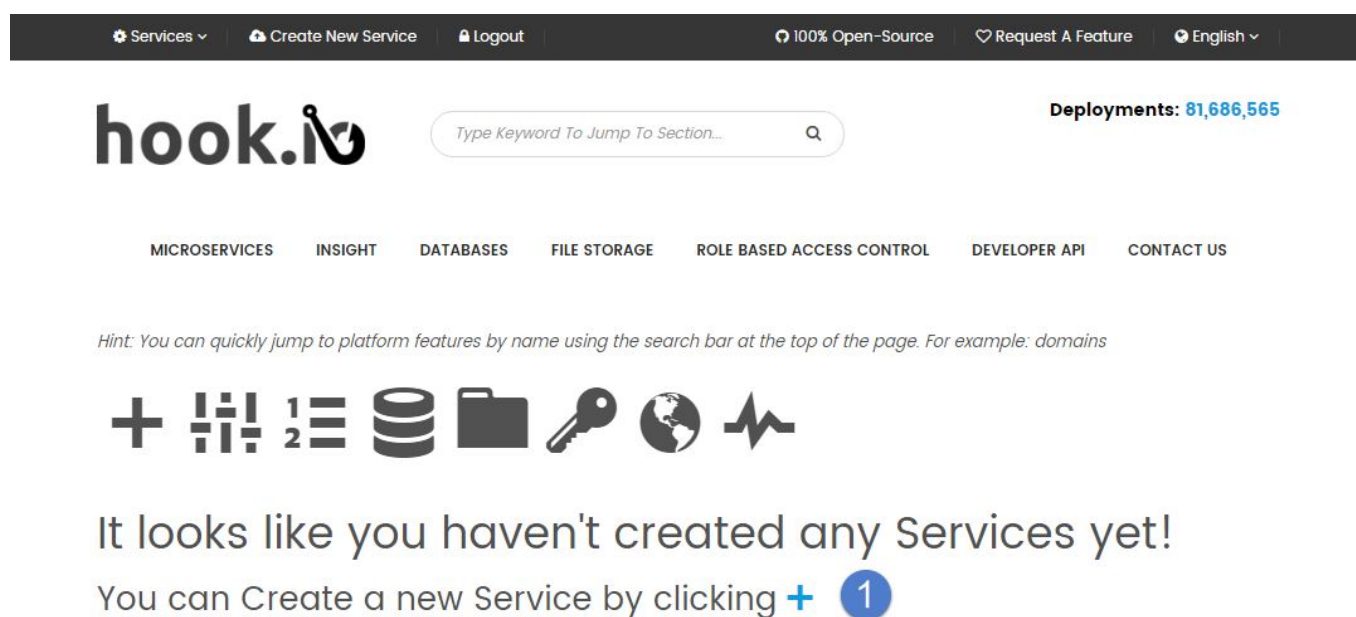
1. Реализация Web-сервиса

В зависимости от назначения, Web-сервис может быть реализован разными способами, с помощью различных языков программирования и на разных платформах.

Исходя из условий кейса, необходим простейший, так называемый, [микросервис](#). Для его реализации можно использовать одну из площадок, предоставляющую возможность создания микросервисов, например open-source проект <https://hook.io>.

После регистрации пользователя площадки и входа в систему отобразится страница управления микросервисами, на которой можно создать новый Web-сервис (рис. 1, 1).

Рис. 1. — Страница управления микросервисами hook.io



После создания нового сервиса необходимо на странице редактирования его свойств задать имя конечной точки доступа к сервису (рис. 2), выбрать язык программирования, на котором будет написан исходный код сервиса (рис. 3, 1), и добавить исходный код сервиса (рис. 3, 2).

Рис. 2. — Создание конечной точки

The screenshot shows the 'Endpoint' configuration form. The title 'Endpoint' is in bold. Below it is the label 'Name' and a text input field containing the value 'bpmonlinehello'. Below the input field is a note: 'Will be part of the url to access the Service. Cannot contain non-url safe characters.'

Рис. 3. — Добавление исходного кода микросервиса


Source Code

Language **1**

Select where the service's source code will come from.

Code Editor Github Gist Github Repo

Github sources are not available unless hook.io is authorized to read from your Github account.

 [Connect Github Account](#)

Using the Code Editor, source is stored directly on hook.io

3

```

1 // A simple hello world microservice
2 // Click "Save Code" to deploy this code
3 // Service will respond to HTTP requests with a string
4 module['exports'] = function helloWorld (hook) {
5   var result = "Добро пожаловать в микросервис bpmonlinehello, " + hook.params.name + "!";
6   hook.res.end(result);
7 };

```

2

После сохранения исходного кода (рис. 3, 3) сервис становится сразу же доступен для использования. Адрес конечной точки <https://hook.io/r-simuta-terrasoft-ru/bpmonlinehello>.

Исходный код сервиса представляет собой функцию, которая на основе параметра *name* HTTP-запроса формирует строку приветствия и отправляет ее в качестве ответа клиентскому приложению. Полностью исходный код микросервиса:

```

module["exports"] = function helloWorld (hook) {
  // Формирование строки ответа.
  var result = "Добро пожаловать в микросервис bpmonlinehello, " + hook.params.name + "!";
  // Отправка ответа клиенту.
  hook.res.end(result);
};

```

К СВЕДЕНИЮ

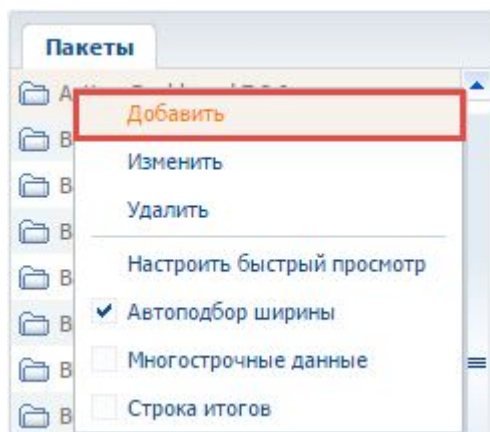
Приведенный исходный код микросервиса работает как с методом POST, так и с методом GET принимаемого HTTP-запроса. Например, если в адресной строке браузера ввести <https://hook.io/r-simuta-terrasoft-ru/bpmonlinehello?name=User>, то сервис обработает корректно.

2. Создание пользовательского пакета

Создание и редактирование пользовательского пакета bpm'online выполняется в разделе [Конфигурация]. Подробно процесс создания пользовательского пакета описан в статье "[Создание пользовательского пакета](#)".

Чтобы создать новый пользовательский пакет, необходимо в контекстном меню на панели [Пакеты] раздела [Конфигурация] выбрать действие [Добавить] (рис. 4).

Рис. 4. — Добавление пользовательского пакета



В результате откроется карточка пакета, в которой необходимо заполнить основные свойства пакета (рис. 5).

Рис. 5. — Основные свойства пакета

Название	tsaHelloApp
Позиция	0
Описание	Пакет приложения AcademyHelloApp

Основные свойства пакета:

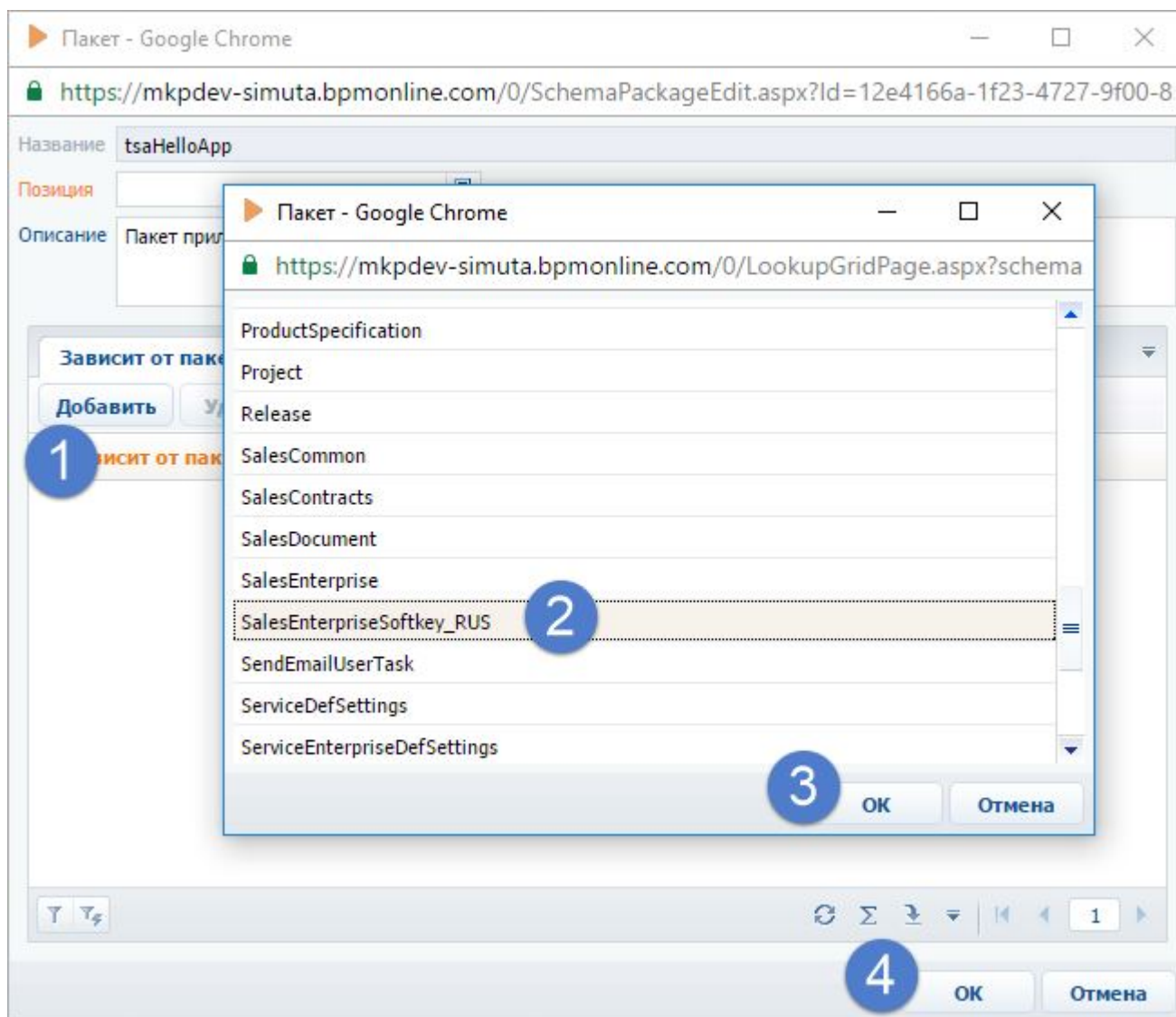
- [Название] — название пакета. Обязательное для заполнения поле. Не может совпадать с названием уже существующих пакетов. В названии обязательно должен быть отображен префикс разработчика, указанный в профиле разработчика.
- [Описание] — описание пакета, например, расширенная информация о функциональности, которая реализуется в пакете. Необязательное поле.

После заполнения полей карточки и нажатия на кнопку [OK] пакет будет создан и отобразится на вкладке [Пакеты].

Для того чтобы созданный пакет имел всю функциональность, которая заложена в систему, для него нужно определить зависимости. При этом достаточно указать самый последний пакет в иерархии предустановленных пакетов. Чтобы определить, какой из пакетов в иерархии предустановленных пакетов является последним, нужно перейти на вкладку [Зависимости пакетов] раздела [Конфигурация]. Далее необходимо найти первый пакет, который находится выше пакета [Custom]. Подробнее об основных пакетах системы и зависимостях пакетов рассказывается в статье "[Зависимости пакетов. Основные пакеты системы](#)".

Для добавления зависимости необходимо в карточке пакета на вкладке [Зависит от пакетов], нажать на кнопку [Добавить] (рис. 6, 1). Затем в справочнике пакетов выбрать необходимый пакет (рис. 6, 2) и нажать на кнопку [OK] (рис. 6, 3). После добавления всех нужных зависимостей пакет необходимо сохранить (рис. 6, 4).

Рис. 6. — Добавление зависимости пакета



3. Создание замещающей схемы

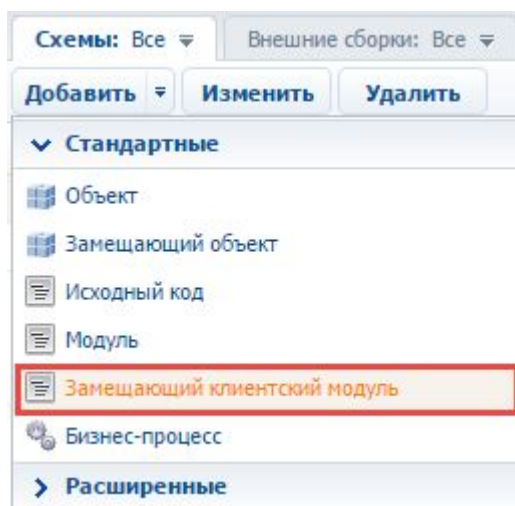
Главное меню приложения bpm'online формируется в нескольких схемах модели представления, образующих иерархию наследования:

- Базовая схема главной страницы (*BaseIntroPageSchema*, пакет *UIv2*) — родительская схема модели представления, в которой формируются основные контейнеры главной страницы. Добавляет в модель представления разделение на левую и правую части, панели отображения видео и панель ссылок на страницы социальных сетей.
- Схема главного меню для базового продукта (*SimpleIntro*, пакет *UIv2*) — наследуется от *BaseIntroPageSchema*. Добавляет на главную страницу ссылки блоков [Базис], [Аналитика] и [Настройка].
- Схема главного меню для продукта Enterprise (*EnterpriseIntro*, пакет *SalesEnterprise*) — наследуется от *SimpleIntro*. Добавляет в левую часть меню дополнительные ссылки секции [Базис], а также ссылки секции [Продажи]. Дополнительно переопределяет ссылку на видеоролик.

Для внесения пользовательских изменений в главное меню продукта Sales Enterprise необходимо заместить схему *EnterpriseIntro* и реализовать в ней дополнительную функциональность. Создание замещающей клиентской схемы подробно рассмотрено в статье "[Создание клиентской схемы](#)".

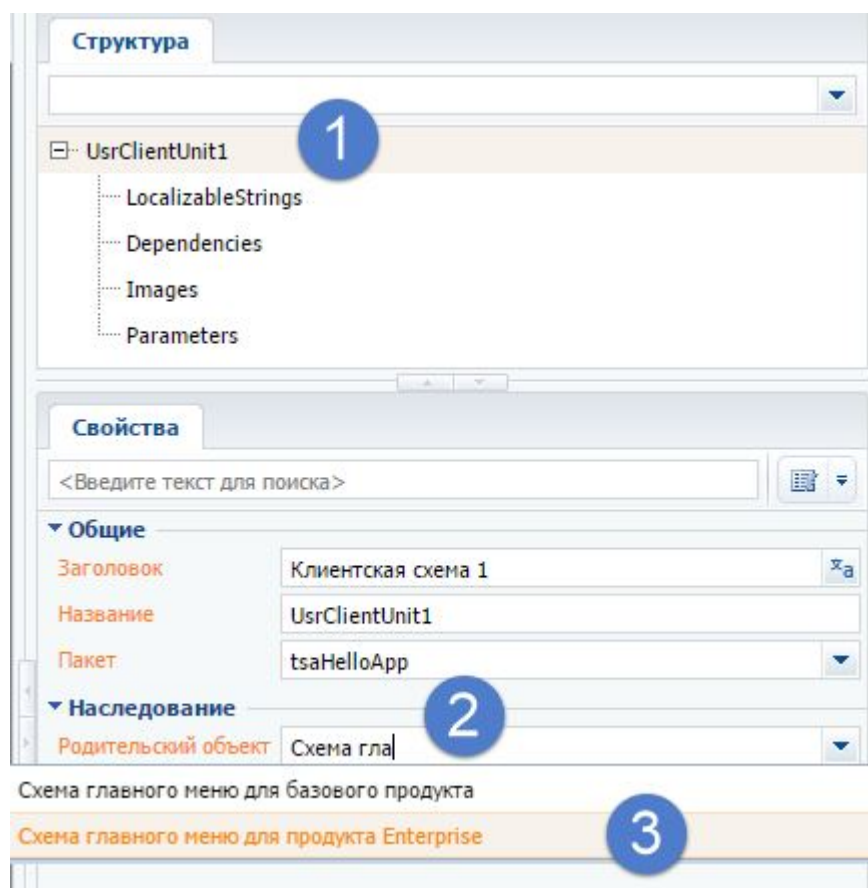
Чтобы создать замещающую схему, необходимо на панели [Пакеты] в разделе [Конфигурация] выбрать созданный на предыдущем шаге пакет. Затем на вкладке [Схемы] выполнить команду [Добавить] > [Замещающий клиентский модуль] (рис. 7).

Рис. 7. — Добавление замещающего клиентского модуля



В появившемся окне дизайнера клиентской схемы необходимо на вкладке [Структура] выбрать корневой элемент структуры схемы (рис. 8, 1), затем на вкладке [Свойства] ввести заголовок замещаемой схемы в поле [Родительский объект]. Во время набора заголовка схемы (рис. 8, 2) система отфильтрует и предложит возможные варианты выбора заголовков (рис. 8, 3). Следует выбрать значение "Схема главного меню для продукта Enterprise".

Рис. 8. — Выбор родительской схемы



После подтверждения выбора родительской схемы в выбранном пользовательском пакете будет создана замещающая клиентская схема модели представления главного меню.

4. Реализация функциональности

Для реализации пользовательской функциональности в созданную схему необходимо добавить исходный код. Подробности о структуре исходного кода схемы модели представления и ее свойствах можно узнать из раздела "[Клиентские схемы модели представления](#)" документации по разработке bpm'online.

Чтобы отобразить результат вызова пользовательского микросервиса на странице главного меню, в схему модели представления в массив модификаций *diff* необходимо добавить конфигурационные объекты для контейнера и содержащейся в контейнере надписи.

Для реализации необходимой пользовательской логики необходимо разработать метод вызова микросервиса *SetHelloAttribute()*, в котором будет использован JavaScript-объект *XMLHttpRequest*. Этот метод необходимо вызвать при загрузке страницы. Для этого необходимо переопределить метод родительской схемы *Init()*, в котором вызвать *SetHelloAttribute()*. Результат вызова запроса к микросервису можно получить в Callback-функции, присвоенной свойству *onreadystatechange* объекта *XMLHttpRequest*. Однако вследствие асинхронности выполнения запроса, результат необходимо сохранить в атрибуте *HelloAttribute* схемы. Сам же атрибут необходимо связать со свойством *caption* надписи, отображающей запрос.

Исходный код схемы модели представления главного меню:

```
define("EnterpriseIntro", [], function() {
    return {
        attributes: {
            // Атрибут, содержащий сообщение от пользовательского сервиса.
            "HelloAttribute": {
                // Тип данных атрибута.
                "dataValueType": Terrasoft.DataValueType.TEXT,
                // Тип атрибута — виртуальная колонка.
                "type": Terrasoft.ViewModelColumnType.VIRTUAL_COLUMN,
                // Значение по умолчанию.
                "value": ""
            }
        },
        methods: {
            // Метод инициализации схемы.
            init: function() {
                // Вызов базовой функциональности.
                this.callParent(arguments);
                // Вызов пользовательского сервиса.
                this.SetHelloAttribute();
            },
            // Метод вызова пользовательского сервиса.
            SetHelloAttribute: function() {
                // Создание HTTP-запроса.
                var xhr = new XMLHttpRequest();
                // URL вызова пользовательского сервиса.
                var url = "https://hook.io/r-simuta-terrasoft-ru/bpmonlinehello";
                // Определение имени текущего пользователя.
                var currUser = Terrasoft.SysValue.CURRENT_USER_CONTACT.displayValue;
                // Формирование параметров HTTP-запроса.
                var params = "name=" + currUser;
                xhr.open("POST", url, true);
                xhr.setRequestHeader("Content-type", "application/x-www-form-
urlencoded");
                // Сохранение контекста.
                var self = this;
                // Callback-функция, вызываемая при получении ответа.
                xhr.onreadystatechange = function() {
                    // Если получен ответ с необходимым статусом.
                    if (xhr.readyState === 4 && xhr.status === 200) {
                        // Установка значения атрибута.
                        self.set("HelloAttribute", xhr.responseText);
                    }
                    else {
                        self.set("HelloAttribute", "Микросервис bpmonlinehello
недоступен!");
                    }
                }
            }
        }
    };
});
```

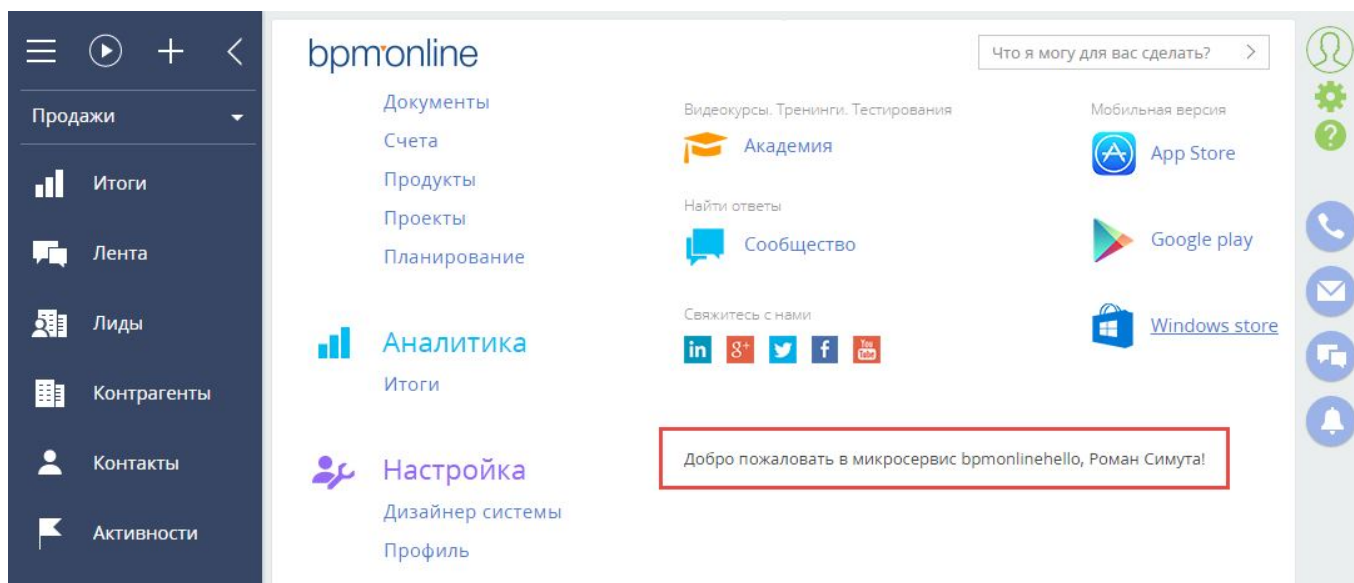
```

        };
        // Отправка HTTP-запроса.
        xhr.send(params);
    }
},
diff: [
    // Пользовательский контейнер, в котором будет содержаться приветствие.
    {
        // Операция добавления.
        "operation": "insert",
        // Имя элемента.
        "name": "HelloContainer",
        // Имя родительского элемента-контейнера.
        "parentName": "RightContainer",
        // Имя свойства родительского элемента для вложенных элементов.
        "propertyName": "items",
        // Конфигурационный объект свойств элемента.
        "values": {
            // Тип элемента – контейнер.
            "itemType": Terrasoft.ViewItemType.CONTAINER,
            // Массив вложенных элементов.
            "items": []
        }
    },
    // Надпись для отображения приветствия.
    {
        "operation": "insert",
        "name": "HelloLabel",
        "parentName": "HelloContainer",
        "propertyName": "items",
        "values": {
            // Тип элемента – надпись.
            "itemType": this.Terrasoft.ViewItemType.LABEL,
            // Ссылка на атрибут с текстом надписи.
            "caption": { "bindTo": "HelloAttribute" }
        }
    }
]
];
});

```

После сохранения схемы необходимо обновить в браузере страницу главного меню приложения. На ней отобразится результат запроса к микросервису `brmonlinehello` (рис. 9).

Рис. 9. — Результат выполнения кейса



1.4 Перенос решения на тестовый сайт

Общие сведения

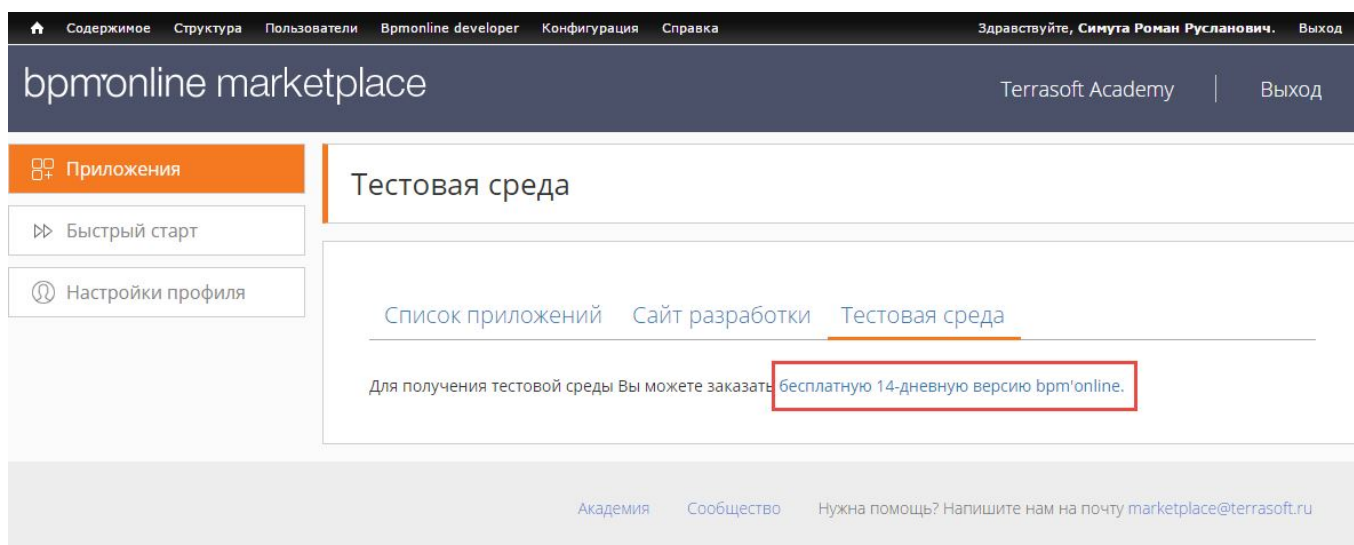
Разработка сложной функциональности требует правильной организации процесса разработки. Рекомендуемый вариант организации процесса разработки подробно изложен в статье "[Организация процесса разработки](#)". Последовательность шагов разработки в случае размещения рабочих сред в облаке приведена в статье "[Рекомендуемая последовательность разработки](#)".

Для тестирования разработанной функциональности рекомендуется использовать бесплатную 14-дневную версию bpm'online.

Регистрация тестового сайта

Заказать бесплатную 14-дневную версию bpm'online для использования в качестве тестовой среды можно непосредственно из Личного кабинета. Для этого необходимо в разделе [Приложения] перейти на вкладку [Тестовая среда] и воспользоваться ссылкой (рис. 1) на страницу заказа пробной версии bpm'online.

Рис. 1. — Ссылка на страницу заказа пробной версии



После перехода по ссылке на странице заказа необходимо заполнить форму (рис. 2) и нажать на кнопку [Приступить к работе].

Рис. 2. — Форма заказа бесплатной версии

bpm'online

БЕСПЛАТНАЯ ВЕРСИЯ

Неограниченные возможности CRM на 14 дней

Попробуйте бесплатную 14-дневную версию CRM-системы с неограниченным количеством пользователей. По истечении тестового периода вы можете оформить подписку и продолжить работу с bpm'online без потери внесенных данных и выполненных настроек.

Бесплатная 14-дневная версия

Фамилия Имя*

Создайте пароль*

Компания*

Подразделение* ▼

Телефон*

Email*

Город*

Страна* ▼

Наполнить систему демо данными

Приступить к работе

[Уже есть бесплатная версия](#)

Тестовое приложение bpm'online сразу же готово к использованию, а ссылка на него с датами активации и деактивации доступна на вкладке [Тестовая среда] (рис. 3).

Рис. 3. — Ссылка на приложение тестовой среды

Перенос решения в тестовую среду

Прежде чем публиковать разработанное приложение в bpm'online marketplace, необходимо удостовериться в его работоспособности на тестовом сайте.

Поскольку сайт разработки и тестовый сайт размещены в облаке bpm'online, разработчики не имеют прямого доступа к возможностям [переноса пакетов между средами](#) с помощью утилиты WorkspaceConsole или SVN. Все эти возможности доступны только через службу поддержки bpm'online.

ВАЖНО

Для приложений, размещенных в облаке, возможен перенос изменений при помощи экспорта и импорта схем. Однако эту возможность использовать не рекомендуется, поскольку решение для marketplace поставляется только в виде пакетов.

Для переноса изменений с сайта разработки в тестовую среду необходимо обратиться в службу поддержки, например, по Email: support@terrasoft.ru. В письме следует указать

- имена пакетов, которые нужно перенести,
- адреса сайта разработки и тестового сайта,
- запрос на получение копий пакетов для резервирования изменений.

Ниже приведен пример запроса в службу поддержки.

Здравствуйете .

Прошу перенести пакет tsaHelloApp из сайта разработки mkpdev-simuta.bpmonline.com на тестовый сайт 003140-crm-bundle.bpmonline.com.

Копию пакета прошу выслать мне .

Спасибо .

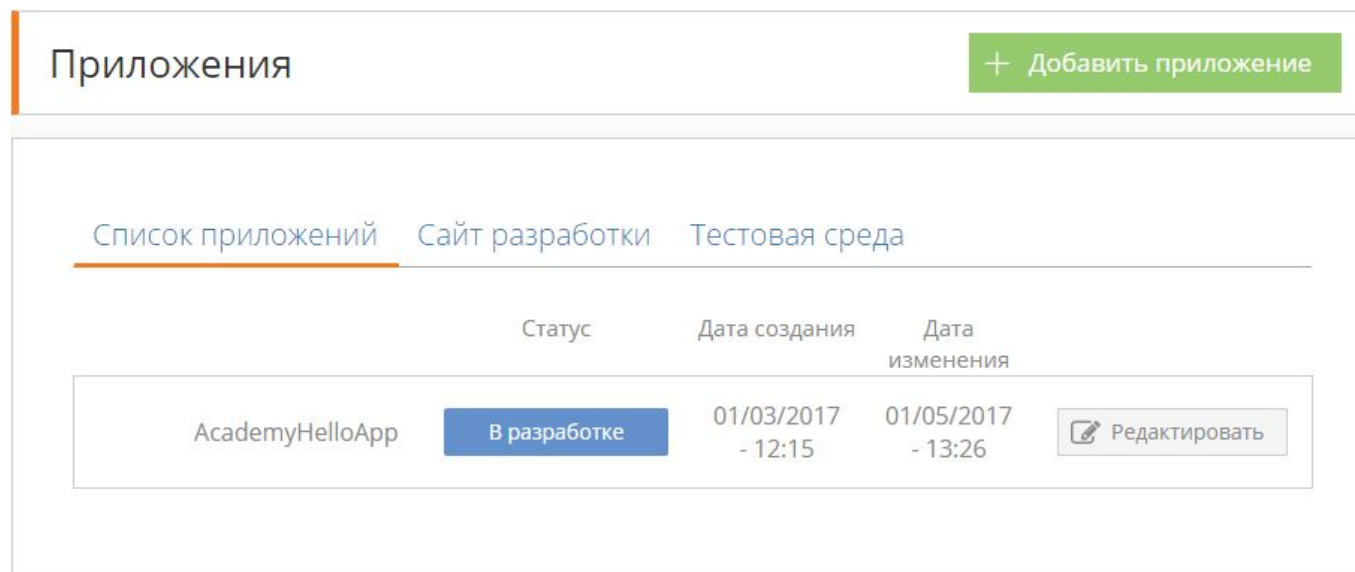
После переноса пакетов в тестовую среду необходимо проверить работоспособность разработанной функциональности. В случае обнаружения ошибок в работе, их необходимо устранить в среде разработки и повторить процедуру переноса пакетов.

1.5 Регистрация решения в Личном кабинете

Зарегистрировать новое решение в Личном кабинете можно на любом этапе разработки. Для этого необходимо перейти в раздел [Приложения] и нажать на кнопку [Добавить приложение]. После

регистрации приложение появится в списке приложений раздела [Приложения] (рис. 1).

Рис. 1. — Список приложений в Личном кабинете

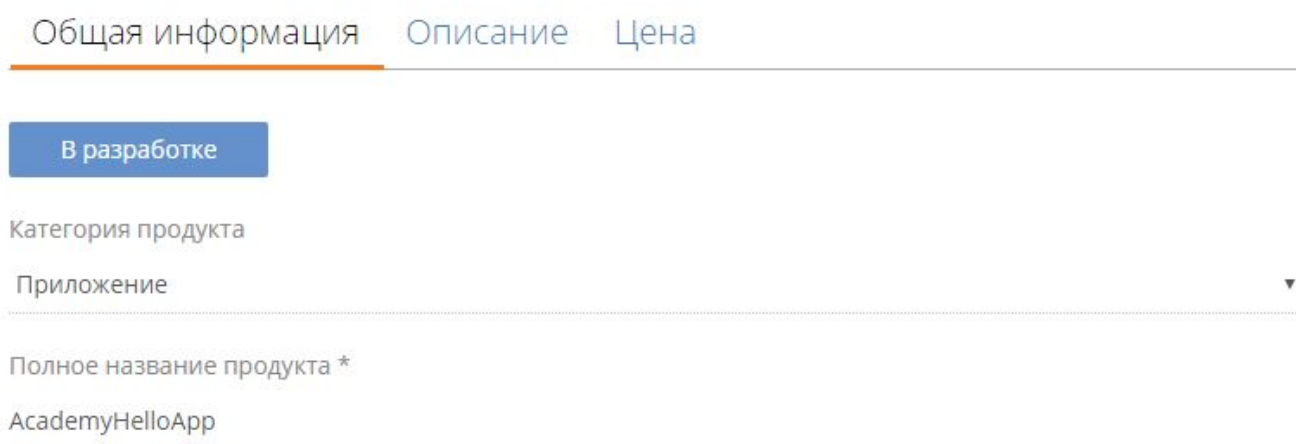


Страница редактирования приложения содержит несколько вкладок, на которых сгруппированы основные свойства приложения. Любые изменения, внесенные на странице редактирования свойств приложения, доступны только для внутреннего использования и не видны посетителям marketplace до момента публикации.

Общая информация

На вкладке [Общая информация] (рис. 2) размещены поля для заполнения основных свойств разрабатываемого приложения bpm'online marketplace.

Рис. 2. — Вкладка [Общая информация]



Свойства приложения, которые необходимо заполнить на этой вкладке:

Статус — стадия жизненного цикла партнерского решения. Решение может обладать одним из статусов:

- В разработке — начальный статус любого партнерского решения. Продукт находится в разработке и недоступен посетителям marketplace.
- На верификации — продукт отправлен на публикацию разработчиком и находится на рассмотрении службы поддержки marketplace. Один из этапов процесса верификации — проверка на совместимость. Заявленная совместимость приложения должна быть реализована и протестирована разработчиком.
- Требуется уточнений — продукт рассмотрен службой поддержки marketplace. Необходимо внести

изменения в финальный вариант продукта согласно общим требованиям к продуктам marketplace и комментариям службы поддержки marketplace.

- Опубликован — продукт верифицирован и опубликован в витрине решений marketplace.



К СВЕДЕНИЮ

Статус устанавливается службой поддержки marketplace и отображается только после регистрации продукта.

Категория продукта — тип категории предоставляемого решения. Возможны два типа решений:

- Вертикальное решение — партнерская конфигурация, разработанная на базе продуктов bpm'online, закрывающая потребность конкретной индустрии и имеющая самостоятельную бизнес-ценность. *Вертикальное решение является самостоятельным продуктом и не требует наличия у пользователя заранее предустановленного продукта bpm'online.*
- Приложение - готовые пакеты расширений, дополняющие функциональность продукта. Приложениями могут быть любые расширения конфигурации (новые объекты, справочники, разделы и их наполнение), готовые бизнес-процессы, виджеты, интеграции, коннекторы и др. *Для использования приложения конечный пользователь должен иметь как минимум одну лицензию продукта bpm'online.*

Полное название продукта — название, под которым решение будет опубликовано в marketplace.

Детальнее о том как правильно назвать продукт для публикации на marketplace можно узнать в регламенте выпуска партнерских решений. Это обязательное свойство.

Совместимость — скрываемая группа полей, которая содержит варианты выбора продуктов bpm'online, с которыми совместимо разрабатываемое приложение, а также версии этих продуктов. Также доступно поле, в котором можно добавить пользовательский комментарий о необходимости установки других решений.

Рис. 3. — Группа полей свойств совместимости приложения

Совместимость ▲

Совместимость (продукт)

- Все продукты на платформе bpm'online
- bpm'online sales
 - bpm'online sales team
 - bpm'online sales commerce
 - bpm'online sales enterprise
- bpm'online service
 - bpm'online: service enterprise
 - bpm'online: customer center
- bpm'online marketing
- bpm'online bank sales
- bpm'online bank customer journey

Совместимость (версия)

7.9 и выше ▼

Совместимость (комментарий)

Format ▾
B *I*
☰ ☱
🔗
🗉
🖼️

Тип развертывания — возможные варианты развертывания приложения. Доступно два варианта выбора (рис. 4):

- Cloud — продукт доступен для развертывания в облаке, в дата-центре Terrasoft.
- On-Site — продукт доступен для установки на собственных серверах клиента.

Рис. 4. — Другие общие свойства приложения

Тип развертывания ▲

- Cloud
- On-site

Локализация

✕ Русский

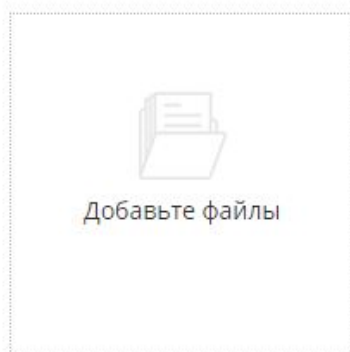
Ссылка на демо

Пакеты

+  tsahelloapp.gz

2.09 КБ

✕



[? Подробнее](#)

Опубликовать

✓ Сохранить

Локализация — перечень языков, на которые локализовано разработанное приложение. Возможен выбор нескольких языков.

Ссылка на демо — ссылка на демо-версию приложения bpm'online с установленным разработанным решением.

Пакеты — все пакеты разрабатываемого решения. Пакеты можно получить, обратившись в службу поддержки.

Описание

На вкладке [Описание] (рис. 2) размещены поля для заполнения основных свойств разрабатываемого приложения bpm'online marketplace.

Рис. 5. — Вкладка [Описание]

Общая информация Описание Цена Модерация *

Краткое описание

Приложение, отображающее приветствие текущего пользователя, полученное из внешнего Web-сервиса.

Детальное описание

Format -



Приложение, отображающее приветствие текущего пользователя, полученное из внешнего Web-сервиса. В качестве внешнего Web-сервиса используется микросервис, разработанный на основе платформы hook.io. В качестве параметра запроса сервису передается имя текущего пользователя приложения bpm'online.

Логотип



Скриншоты



Свойства приложения, которые необходимо заполнить на этой вкладке:

Краткое описание — краткое описание предоставляемого продукта. Его основная функциональность, решаемые задачи. Ограничение — 115 символов, включая пробелы.

Детальное описание — более детальное описание предоставляемого продукта.

Логотип — логотип продукта, отображаемый в витрине marketplace. Изображение корпоративного фона в формате .png, .gif, .jpg или .jpeg с разрешением 262 px по ширине и 216 px по высоте. Не белого цвета, желательно, в темных тонах. В нижней половине изображения необходимо отобразить логотип разработчика белым цветом.

Скриншоты — изображения с копиями экрана работающего приложения. Допустимые форматы — .png, .gif, .jpg или .jpeg. Минимальное разрешение 1024 px по ширине. Максимальный размер файла 20 МБ.

Ссылка на видео — ссылка на видеоролик, в котором описывается работа приложения.

Условия поддержки — описание условий предоставления технической поддержки. Разработчик обязуется обеспечить техническую поддержку пользователей в рамках реализованной функциональности по электронной почте, телефону или любому другому каналу связи.

Цена

На вкладке [Описание] (рис. 2) размещены поля для заполнения основных свойств разрабатываемого приложения bpm'online marketplace.

Рис. 6. — Вкладка [Описание]

Свойства приложения, которые необходимо заполнить на этой вкладке:

Тип развертывания — краткое описание типа развертывания предоставляемого продукта. Например, bpm'online sales enterprise cloud.

Цена — количественное выражение стоимости продукта в зависимости от формата. Если поле [Цена] остается незаполненным, это подразумевает, что продукт в указанном формате бесплатен.

Формат — ценовая модель для предоставляемого продукта. Основные модели ценообразования платных решений:

- Год — стоимость использования продукта за год.
- Единоразово — клиент оплачивает указанную сумму один раз и может пользоваться продуктом неограниченный срок.
- За пользователя — указанная цена умножается на количество именных пользователей продукта. Клиент может пользоваться продуктом неограниченный срок.
- За пользователя / год — указанная цена умножается на количество именных пользователей продукта. Клиент оплачивает подписку сроком на один год и через год должен продлить подписку.
- По запросу — стоимость устанавливается по запросу.

Комментарий — комментарий разработчика по поводу его ценовой политики.

1.6 Публикация решения в marketplace

Общие сведения

После завершения процесса разработки решения и его успешного тестирования готовый продукт может быть опубликован в витрине marketplace. Однако, прежде чем публиковать продукт, рекомендуется ознакомиться с [регламентом выпуска партнерских решений](#). Кроме того, предоставляемое решение должно соответствовать общим требованиям к продуктам marketplace.

Общие требования к продуктам marketplace

Основные требования к материалам для публикации:

1. Короткое описание продукта — краткое, максимально привлекательное для пользователей описание решения. Должно включать описание основной функциональности и решаемой задачи. Ограничение — 115 символов с пробелами.
2. Детальное описание продукта — полное описание возможностей и преимуществ вашего решения.
3. Скриншоты — как минимум 1 скриншот, описывающий функциональность решения. Минимальное разрешение 1024 px по ширине.
4. Совместимость — указание продуктов и версий bpm'online, с которыми совместимо данное решение.
5. Цена — указание ценовой модели для данного решения.
6. Пакеты — пакеты с реализованной функциональностью, которая заявлена в описании продукта.
7. Логотип приложения — изображение корпоративного фона в формате .png, .gif, .jpg или .jpeg с разрешением 262 px по ширине и 216 px по высоте. Не белого цвета, желательно в темных тонах. В нижней половине изображения необходимо отобразить логотип разработчика белым цветом.
8. Логотип разработчика — изображение в формате .png, .gif, .jpg или .jpeg с разрешением 200 px по ширине, предпочтительно на белом фоне.

Требования к приложению:

1. Работоспособность — решение должно работать именно так, как было заявлено в описании.
2. Совместимость — решение должно быть совместимо с теми продуктами и версиями bpm'online, которые были заявлены при регистрации.
3. Производительность — решение не должно приводить к снижению производительности продукта, для которого оно предназначено.
4. Описание — описание продукта не должно содержать лексических, синтаксических или смысловых ошибок.
5. Конфиденциальность данных — приложение не должно незаконно получать доступ и передавать данные из приложения, в которое оно установлено. Любая передача данных, которая происходит после установки (в случае интеграции со сторонними решениями), должна быть явно упомянута в описании и должна происходить только после авторизации пользователя.
6. Поддержка — разработчик обязуется обеспечить техническую поддержку пользователей в рамках реализованной функциональности по электронной почте, телефону или любому другому каналу связи. Условия поддержки должны быть явно описаны в соответствующем пункте страницы свойств приложения.
7. Обновление — при обновлении версий существующего приложения разработчик обязуется указать в описании все исправления и модификации, которые были произведены.

Публикация решения в marketplace



Для публикации разработанного решения необходимо перейти на страницу приложения в Личном кабинете и нажать на кнопку [Опубликовать] (рис. 1).

Рис. 1. — Кнопка [Опубликовать]



После отправки готового продукта на публикацию, он будет автоматически передан в службу поддержки marketplace для последующей верификации. Данный этап будет обозначен в продукте статусом "На верификации" (рис. 2).

Рис. 2. — Изменение статуса приложения

Список приложений		Сайт разработки	Тестовая среда		
	Статус	Дата создания	Дата изменения		
	AcademyHelloApp	На верификации	01/03/2017 - 12:15	01/11/2017 - 11:25	 Редактировать

Верификация решения и его размещение в витрине marketplace

Процесс верификации разработанного приложения для bpm'online marketplace необходим для проверки соответствия решения требованиям к продуктам marketplace.

В процессе верификации у службы поддержки marketplace могут возникнуть замечания к публикуемому продукту. Все замечания будут направлены разработчику на указанный в профиле разработчика Email с подробными комментариями и рекомендациями к исправлению. Сам продукт при этом получит статус "Требуется уточнений". Для завершения публикации необходимо устранить все замечания и повторно опубликовать продукт, нажав на кнопку [Опубликовать].

После того как продукт будет верифицирован службой поддержки marketplace, он будет автоматически размещен в витрине marketplace. Статус приложения в Личном кабинете будет изменен на "Опубликовано". После публикации соответствующее уведомление будет отправлено разработчику по электронной почте.

1.7 Примеры разработки приложений

Разработка функциональности пользовательского решения является одним из шагов общей последовательности разработки приложений для marketplace.

Подробный пример интеграции стороннего пользовательского сервиса в решение для bpm'online marketplace описан в статье "**Пример разработки простого приложения для bpm'online marketplace (Section 1.3)**".

Разработка решения, которое добавляет новый раздел в приложение bpm'online, подробно рассмотрена в статье "**Пример разработки приложения с пользовательским разделом (Section 1.7.1)**".

Процесс переноса демоданных определенного раздела и их привязка к пакету рассмотрены на примере в статье "**Привязка данных к пакету (Section 1.7.2)**".

1.7.1 Пример разработки приложения с пользовательским разделом

Общие сведения

Приложение для bpm'online marketplace, в котором реализован пользовательский раздел, практически ничем не отличается от разработки обычного проектного решения создания пользовательского раздела.

Подробности об основных принципах разработки пользовательских разделов в bpm'online можно узнать из подраздела "[Бизнес-логика раздела](#)" документации по разработке.

Для создания пользовательского раздела необходимо использовать [мастер разделов](#). Все схемы, создаваемые мастером разделов, сохраняются в пакет, который указан в системной настройке [Текущий пакет] (*CurrentPackageId*). По умолчанию текущим пакетом является пакет [Custom].

Однако, если изменения необходимо переносить в другую среду, то вести разработку в пакете [Custom] нельзя. Это связано с тем, что сам пакет [Custom] является системным пакетом, следовательно, его нельзя экспортировать ни с помощью утилиты WorkspaceConsole, ни с помощью SVN. Пакет [Custom] подробно описан в статье "[Пакет \[Custom\]](#)" документации по разработке.

К СВЕДЕНИЮ

Из пакета [Custom] можно экспортировать только схемы с помощью [механизма экспорта и импорта схем](#). Однако привязанные к пакету данные и SQL-сценарии с помощью этого механизма перенести в другой пакет нельзя.

Также в свойстве схемы [Пакет] можно указать любой другой пользовательский пакет, и после сохранения она автоматически будет перенесена в указанный пакет.

Поэтому всю разработку новой функциональности следует вести в пользовательском пакете.

Последовательность создания приложения с пользовательским разделом:

1. Создать пользовательский пакет.
2. Установить значения системных настроек [Текущий пакет] и [Префикс названия объекта] для разработки в пользовательском пакете. Значение системной настройки [Префикс названия объекта] должно соответствовать префиксу, указанному в Кабинете разработчика (см. "[Настройка профиля разработчика \(Section 1.1\)](#)").
3. С помощью [мастера разделов](#) создать раздел.
4. Реализовать необходимую функциональность раздела.
5. Привязать к пользовательскому пакету все необходимые данные, относящиеся к разделу.

Описание приложения

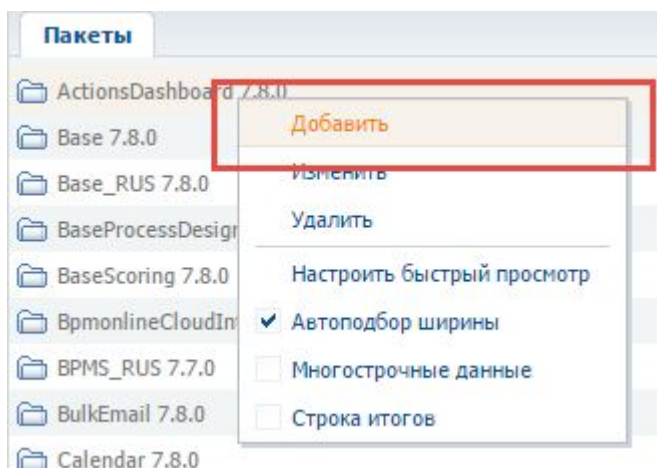
Создать проектное решение, реализующее в рабочем месте [Маркетинг] новый раздел [Web-данные]. На странице редактирования записи раздела нужно реализовать отображение статистики по заданному названию сайта (его URL). Статистику отображать в html-элементе <iframe> с помощью стороннего Web-сайта <https://www.similarweb.com>.

Алгоритм реализации приложения

1. Создание пользовательского пакета

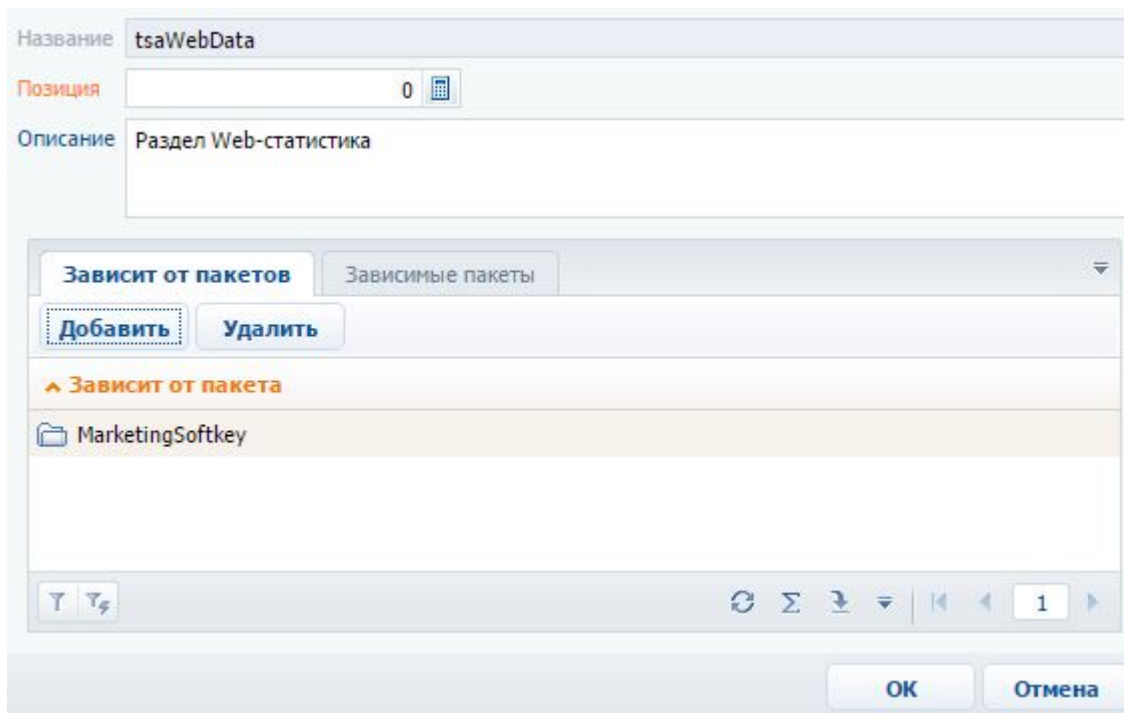
Процесс создания пользовательского пакета подробно описан в статье "[Создание пользовательского пакета](#)" документации по разработке. Чтобы создать новый пользовательский пакет, необходимо в контекстном меню на вкладке [Пакеты] раздела [Конфигурация] выбрать действие [Добавить] (рис. 1).

Рис. 1. — Добавление нового пакета



Затем следует заполнить основные поля открывшейся карточки пакета и сохранить пакет. Открыв карточку пакета повторно, следует добавить зависимости пакета. Поскольку по исходным данным необходимо добавить раздел в рабочее место [Маркетинг], то необходимо добавить зависимость от пакета [MarketingSoftkey] (рис. 2).

Рис. 2. — Карточка пакета



ВАЖНО

Имя пакета должно содержать префикс, указанный в профиле разработчика (см. "**Настройка профиля разработчика (Section 1.1)**").

К СВЕДЕНИЮ

Указанные зависимости пакета будут напрямую влиять на совместимость приложения с тем или иным продуктом линейки bpm'online.

2. Установка значений системных настроек [Текущий пакет] и [Префикс названия объекта]

Для того чтобы мастер разделов сохранил связанные с разделом схемы в пользовательский пакет, следует

указать этот пакет в качестве текущего. Для этого необходимо в дизайнера системы перейти в раздел [Системные настройки], открыть системную настройку [Текущий пакет] и в поле [Значение по умолчанию] выбрать из справочника пакетов необходимый пользовательский пакет (рис. 3).

Рис. 3. — Редактирование системной настройки [Текущий пакет]

Затем системную настройку необходимо сохранить.

Также необходимо установить указанное в **профиле разработчика (Section 1.1)** значение для системной настройки [Префикс названия объекта] (рис. 4).

Рис. 4. — Системная настройка [Префикс названия объекта]

Этот префикс будет автоматически добавлен мастером к названиям всех схем, которые он создаст.

3. Создание раздела с помощью мастера

Процесс создания нового раздела с помощью мастера подробно описан в статьях "[Мастер разделов](#)", "[Как настроить свойства раздела](#)" и "[Как настроить страницу раздела](#)" документации bpm'online, а также в статье "[Создание нового раздела](#)".

Для нового раздела в мастере необходимо указать значения для следующих свойств (рис. 5):

Рис. 5. — Свойства создаваемого раздела в мастере разделов

Web-данные: Основные свойства


СОХРАНИТЬ ОТМЕНА < РАЗДЕЛ СТРАНИЦА >

Укажите основные свойства нового раздела:

Заголовок* Web-данные

Код (на английском)* tsaWebData

Рабочее место Маркетинг

Иконка раздела 

Укажите настройки страницы:

Использовать одну страницу для всех записей

Использовать несколько страниц

- [Заголовок] — "Web-данные". Заголовок отображается в главном меню приложения и на странице раздела.
- [Код] — "tsaWebData". Это название схемы объекта раздела. Подробнее о создаваемых мастером раздела схемах можно узнать в статье "[Создание нового раздела](#)".
- [Рабочее место] — "Маркетинг". Рабочее место, в котором будет отображен раздел.

Исходя из условий примера, на странице редактирования записи раздела необходимо создать поле для ввода названия URL сайта, для которого будет отображена статистика. Для этого можно использовать колонку [Название], создаваемую мастером по умолчанию (рис. 6, 1). Также нужно добавить вспомогательное поле (2), в котором будет храниться создаваемая программно ссылка на страницу статистики Web-сайта <https://www.similarweb.com> для введенного URL. Эта ссылка будет присваиваться атрибуту `src` элемента `<iframe>`, размещенного на странице. Для размещения элемента `<iframe>` необходимо добавить новую вкладку [Web-данные] (3) на панель вкладок.

Рис. 6. — Добавление полей и вкладки на страницу редактирования записи

The screenshot shows a web application interface with a navigation bar at the top containing 'РАЗДЕЛ' and 'СТРАНИЦА'. Below the navigation bar is a form with three numbered fields: 1. 'Название' (Name), 2. 'URL', and 3. 'WEB-ДАННЫЕ' (Web Data). Below the form are two buttons: 'ДОБАВИТЬ ГРУППУ ПОЛЕЙ' (Add Field Group) and 'ДОБАВИТЬ ДЕТАЛЬ' (Add Detail).

После нажатия на кнопку [Сохранить] мастер создаст все необходимые схемы в пакете и сохранит результаты в базу данных.

К СВЕДЕНИЮ

Для завершения работы мастеру разделов может потребоваться несколько минут. После сохранения всех результатов работы мастера, в системе отобразится соответствующее сообщение.

Для отображения нового раздела необходимо перезагрузить страницу браузера с очисткой кеша.

4. Реализация необходимой функциональности раздела

Созданный при помощи мастера раздел уже имеет всю необходимую функциональность по добавлению, удалению и сортировке записей. Также в схеме страницы редактирования уже унаследована вся функциональность, необходимая для сохранения и загрузки основных данных записи.

Для формирования ссылки на страницу статистики, отображаемой в поле URL, необходимо добавить кнопку. Для добавления кнопки используется конфигурационный объект, добавляемый в массив *diff* модели представления страницы. Заголовок кнопки связан с локализуемой строкой *AddUrlButtonCaption*. Событие *click* нажатия кнопки нужно связать с методом *addUrl()*, в котором формируется ссылка на страницу статистики и устанавливается новое значение для колонки *tsaURL*, связанной с полем [URL] страницы редактирования. Также эта ссылка присваивается атрибуту *src* элемента `<iframe>`. Подробнее о создании кнопки на странице редактирования можно узнать, например, из статьи "[Добавление кнопки](#)". Подробнее о массиве *diff* изложено в статье "[Массив diff](#)" документации по разработке.

Элемент `<iframe>` добавляется на созданную мастером вкладку также с помощью конфигурационного объекта, добавляемого в массив *diff*. В свойстве *html* конфигурационного объекта формируется html-элемент `<iframe>` с заданием нужных стилей. При этом атрибут *src* не указывается, поскольку он формируется программно в методе *addUrl()*. Для того чтобы в элементе `<iframe>` данные восстанавливались после переключения на другую вкладку, необходимо связать событие *afterrender* перерисовки контейнера, содержащего элемент `<iframe>`, с методом *addUrl()*.

После открытия сохраненной записи, событие *afterrender* отрисовки контейнера с элементом `<iframe>` наступает, как правило, раньше полной загрузки данных. Поэтому метод *addUrl()* следует также вызвать в методе-обработчике события загрузки данных *onEntityInitialized()*.

Полностью исходный код схемы модели представления страницы редактирования записи:

```
define("tsaWebData1Page", [], function() {
    return {
        entitySchemaName: "tsaWebData",
        details: /**SCHEMA_DETAILS*/{}/**SCHEMA_DETAILS*/,
        diff: /**SCHEMA_DIFF*/[
            // Название Web-страницы.
            {
                "operation": "insert",
```



```

    "name": "tsaName",
    "values": {
      "layout": {
        "colSpan": 24,
        "rowSpan": 1,
        "column": 0,
        "row": 0,
        "layoutName": "ProfileContainer"
      },
      "bindTo": "tsaName"
    },
    "parentName": "ProfileContainer",
    "propertyName": "items",
    "index": 0
  },
  // Ссылка на статистику Web-страницы.
  {
    "operation": "insert",
    "name": "tsaURL",
    "values": {
      "layout": {
        "colSpan": 24,
        "rowSpan": 1,
        "column": 0,
        "row": 0,
        "layoutName": "Header"
      },
      "labelConfig": {},
      "enabled": true,
      "readonly": true,
      "bindTo": "tsaURL"
    },
    "parentName": "Header",
    "propertyName": "items",
    "index": 0
  },
  // Вкладка на панели вкладок.
  {
    "operation": "insert",
    "name": "TabData",
    "values": {
      "caption": "Web-данные",
      "items": []
    },
    "parentName": "Tabs",
    "propertyName": "tabs",
    "index": 0
  },
  // Кнопка добавления URL.
  {
    "operation": "insert",
    "parentName": "ProfileContainer",
    "propertyName": "items",
    "name": "AddUrlButton",
    "values": {
      "layout": {
        "colSpan": 24,
        "rowSpan": 1,
        "column": 0,
        "row": 1
      },
      "itemType": Terrasoft.ViewItemType.BUTTON,

```

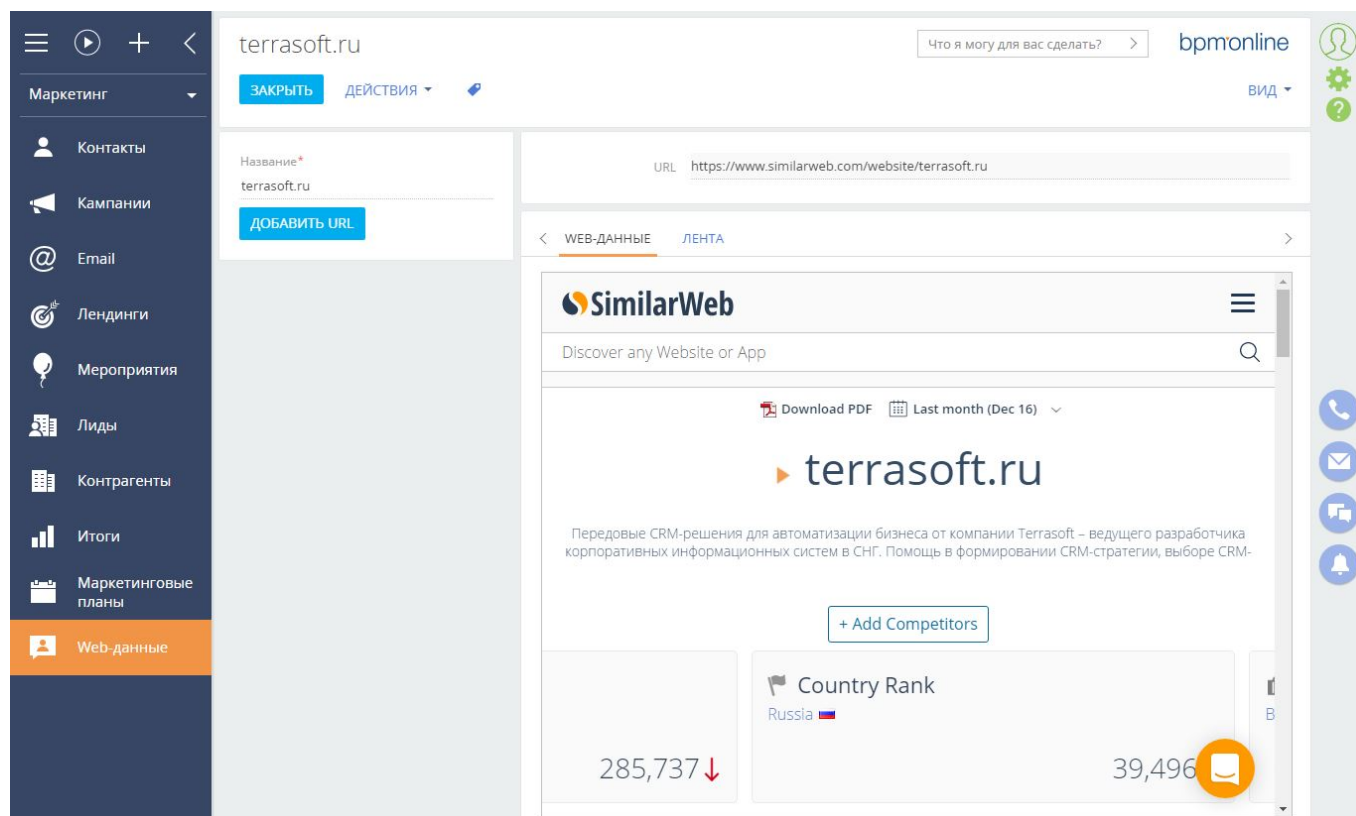
```

        "caption": {"bindTo": "Resources.Strings.AddUrlButtonCaption"},
        "click": {"bindTo": "addUrl"},
        "style": Terrasoft.controls.ButtonEnums.style.BLUE
    }
},
// Контейнер с внедренным Html-элементом iframe.
{
    "operation": "insert",
    "name": "IFrameStat",
    "parentName": "TabData",
    "propertyName": "items",
    "values": {
        "id": "testiframe",
        "itemType": Terrasoft.ViewItemType.CONTAINER,
        "selectors": {"wrapEl": "#stat-iframe"},
        "layout": { "colSpan": 24, "rowSpan": 1, "column": 0, "row": 0 },
        "html": "<iframe id='stat-iframe' class='stat-iframe'
width='100%' height='550px'" +
            "style = 'border: 1px solid silver;'></iframe>",
        "afterrender": {
            "bindTo": "addUrl"
        }
    }
}
]/**SCHEMA_DIFF*/,
methods: {
    // Обработчик события полной загрузки данных.
    onEntityInitialized: function() {
        // Вызов родительской реализации метода.
        this.callParent(arguments);
        // Вызов метода добавления URL к Html-элементу iframe
        this.addUrl();
    },
    // Метод добавления URL к Html-элементу iframe.
    addUrl: function() {
        // Получение компонента по его id.
        var iframe = Ext.get("stat-iframe");
        if (!iframe) {
            window.console.error("Не найдена вкладка с элементом iframe");
            return;
        }
        // Получение значения колонки [tsaName].
        var siteName = this.get("tsaName");
        if (!siteName) {
            window.console.error("Не указано имя сайта");
            return;
        }
        // Формирование ссылки на страницу статистики.
        var url = "https://www.similarweb.com/website/" + siteName;
        // Установка значения колонки [tsaName].
        this.set("tsaURL", url);
        // Присвоение ссылки на страницу статистики Html-элементу iframe.
        iframe.dom.src = url;
    }
},
rules: {}
});
});

```

После сохранения схемы, на странице редактирования записи раздела [Web-статистика] появится кнопка добавления URL и элемент <iframe>, в котором будет отображаться статистика по введенному Web-ресурсу (рис. 7).

Рис. 7. — Страница редактирования записи раздела



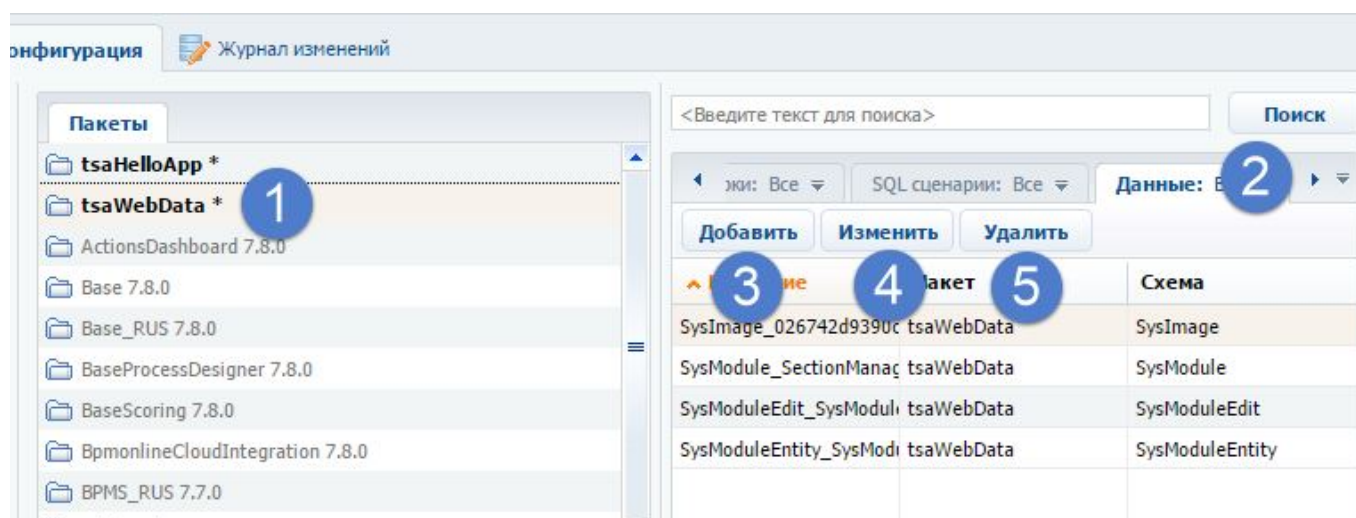
1.7.2 Привязка данных к пакету

Общие сведения

После завершения работы мастера раздела все необходимые для функционирования созданного раздела данные уже привязаны к пользовательскому пакету. Однако, если требуется, чтобы после установки приложения раздел отображался в определенном рабочем месте с некоторым демо-наполнением, к пользовательскому пакету нужно привязать дополнительные данные.

Привязка данных к пакету выполняется в разделе [Конфигурация]. Чтобы посмотреть, какие данные уже привязаны к пакету, необходимо выбрать нужный пакет (рис. 1, 1) и перейти на вкладку [Данные] (2). Привязанные данные можно добавлять (3), изменять (4) и удалять (5).

Рис. 1. — Вкладка [Данные] раздела [Конфигурация]

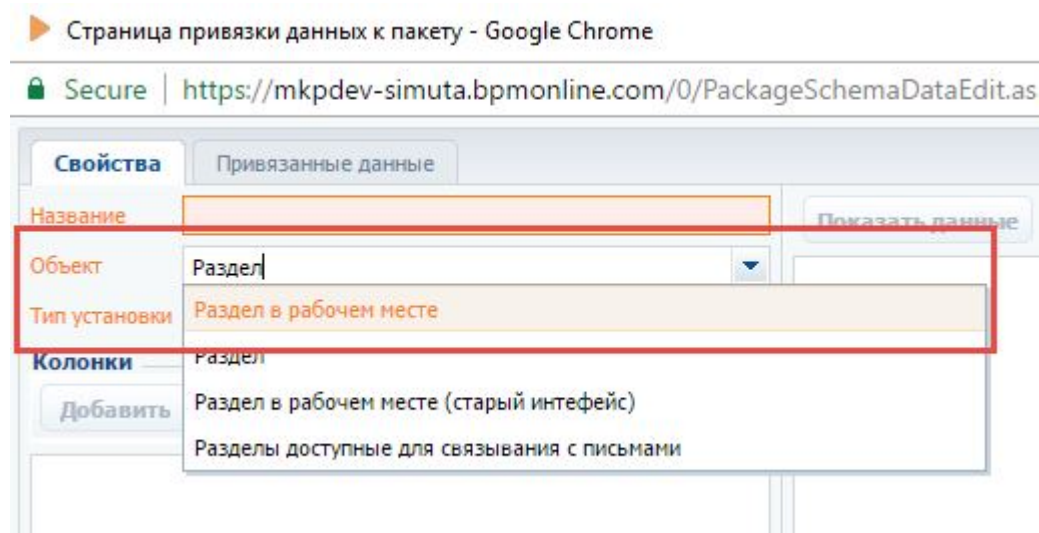


Приведенные ниже примеры привязки данных выполнены на основе раздела, созданного после выполнения последовательности действий, описанной в статье "**Пример разработки приложения с пользовательским разделом (Section 1.7.1)**".

Привязка раздела к рабочему месту

Для отображения раздела в рабочем месте [Маркетинг] нужно привязать данные схемы [Раздел в рабочем месте]. Для этого следует нажать на кнопку [Добавить] и на появившейся странице привязки данных к пакету в поле [Объект] выбрать значение "Раздел в рабочем месте" (рис. 2).

Рис. 2. — Выбор объекта привязываемых данных

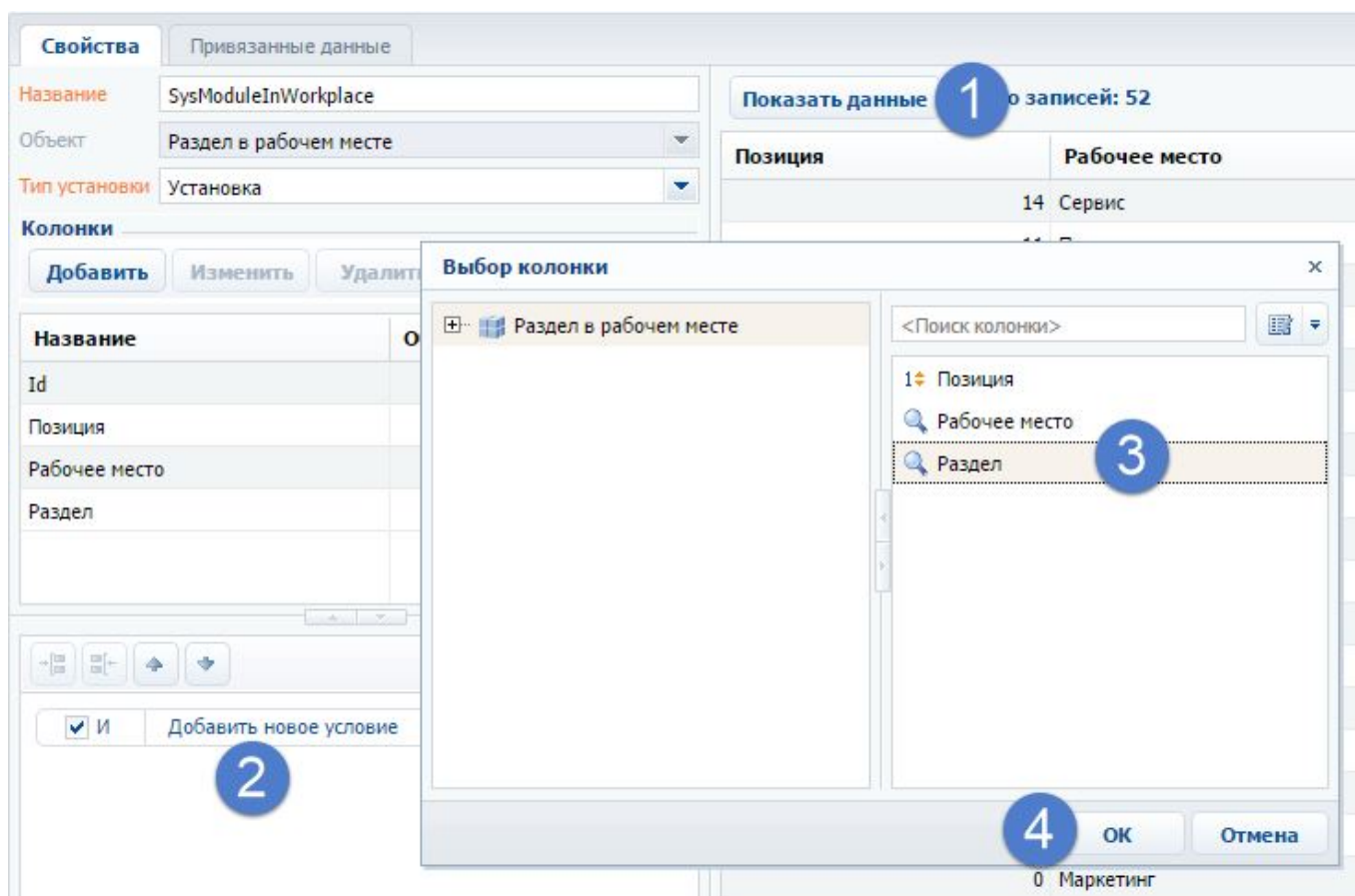


В появившемся диалоговом окне с запросом о сохранении результатов нужно нажать на кнопку [Да].

Затем следует выбрать тип установки. Наиболее универсальный вариант — "Установка". При этом данные будут установлены как при первичной установке пакета, так и при последующих обновлениях.

После нажатия на кнопку [Показать данные] (рис. 3, 1) отобразятся данные обо всех разделах и рабочих местах приложения, в котором ведется разработка.

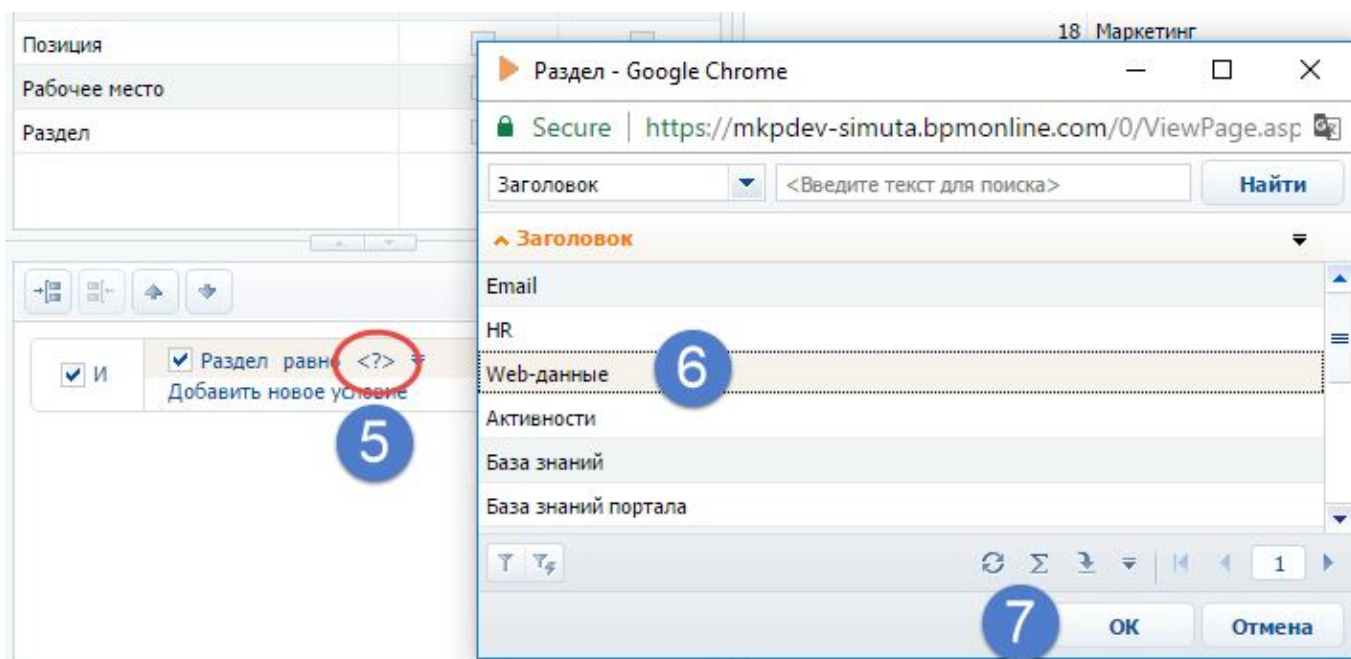
Рис. 3. — Отображение и фильтрация привязываемых данных



Поскольку необходимо привязать данные только по созданному разделу, нужно выполнить фильтрацию привязываемых данных. Для этого в области фильтров нужно добавить новое условие (рис. 3, 2), затем выбрать колонку, по которой будет строиться условие (3) и нажать на кнопку [OK] (4).

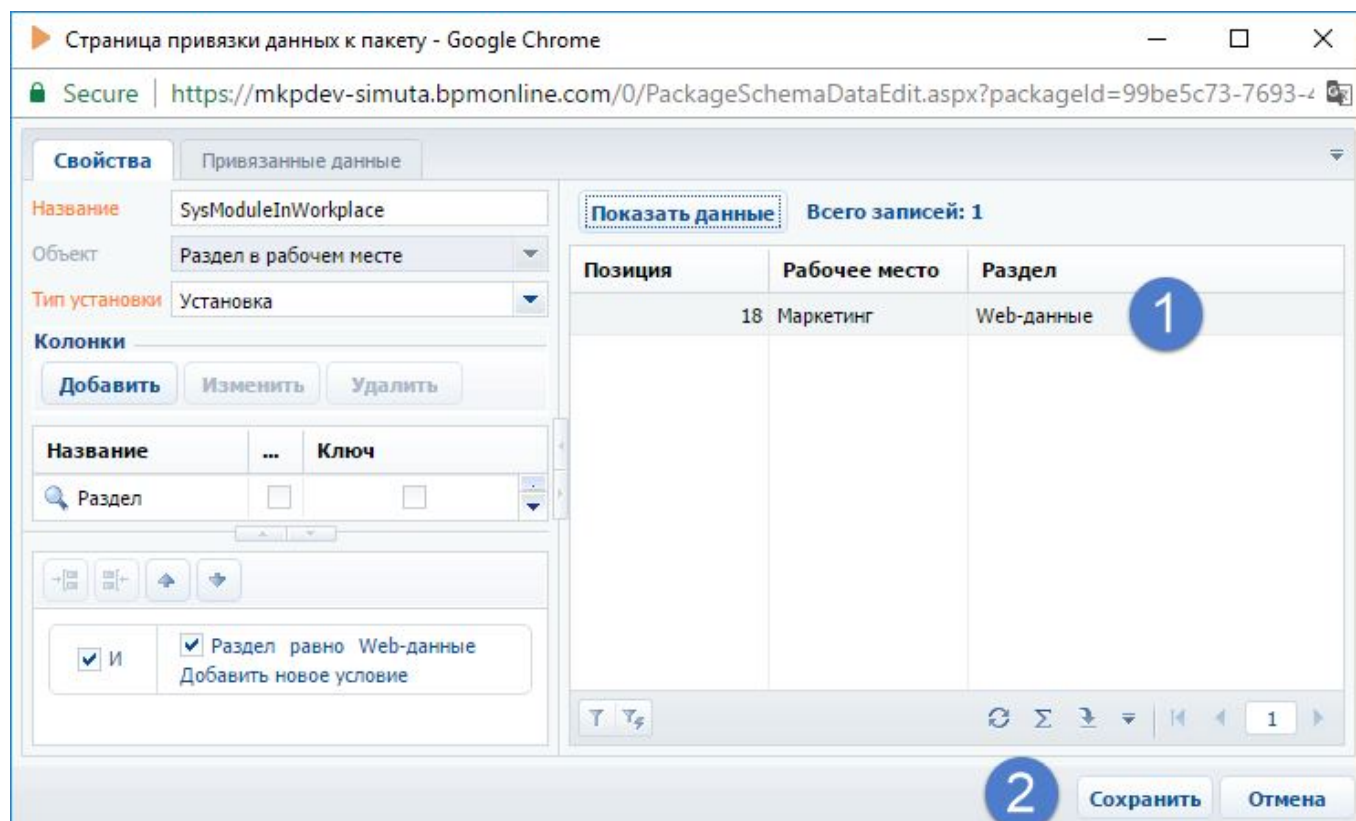
Затем нужно построить вторую часть условия. Для этого необходимо кликнуть на знак условия (рис. 4, 5), после чего выбрать название раздела (6) и нажать на кнопку [OK] (7).

Рис. 4 – Построение условия фильтрации привязываемых данных



В результате будет построен фильтр выбора созданного раздела (рис. 5, 1).

Рис. 5. — Построение условия фильтрации привязываемых данных



После сохранения привязанных данных (рис. 5, 2) в разделе [Конфигурация] на вкладке [Данные] появится новая запись (рис. 6).

Рис. 6. — Привязанные данные раздела в рабочем месте

Схемы: Все	Внешние сборки: Все	SQL сценарии: Все	Данные: Все
<p>Добавить Изменить Удалить</p>			
Название	Пакет	Схема	
SysImage_026742d9390c4778bc469	tsaWebData	SysImage	
SysModule_SectionManager_b9d4f31	tsaWebData	SysModule	
SysModuleEdit_SysModuleEditManag	tsaWebData	SysModuleEdit	
SysModuleEntity_SysModuleEntityMa	tsaWebData	SysModuleEntity	
SysModuleInWorkplace	tsaWebData	SysModuleInWorkplace	

Привязка данных наполнения раздела

Чтобы привязать данные демонаполнения раздела (рис. 7), необходимо сначала определить заголовок схемы основного объекта раздела (рис. 8).

Рис. 7. — Демонополнение раздела

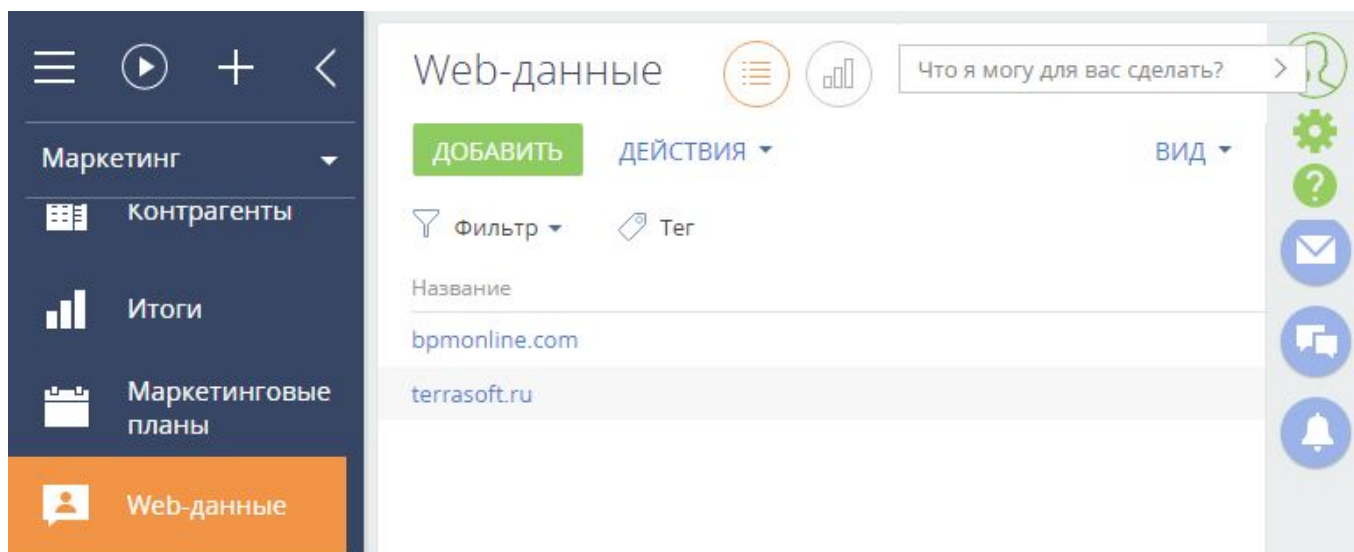
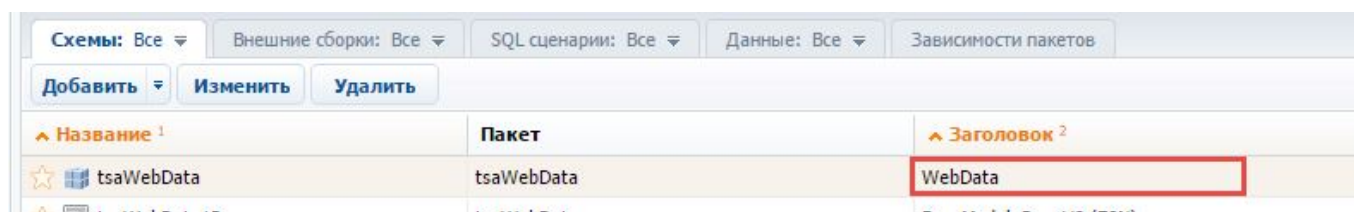


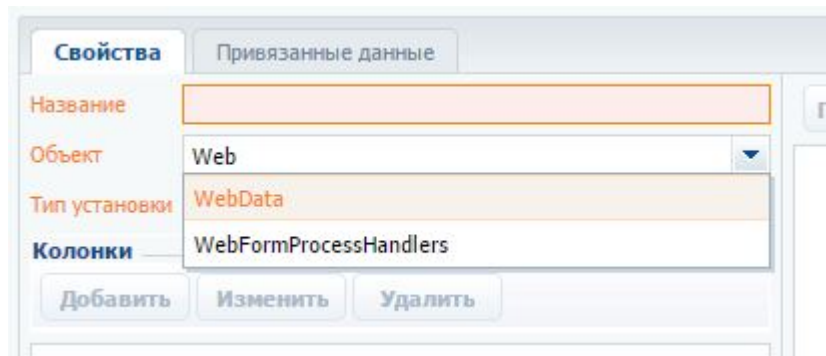
Рис. 8. — Заголовок схемы объекта раздела



Эта схема создается автоматически мастером раздела и ее заголовок должен совпадать с заголовком раздела.

Для привязки данных следует нажать на кнопку [Добавить] во вкладке [Данные] раздела [Конфигурация] и на появившейся странице привязки данных к пакету в поле [Объект] выбрать значение, соответствующее заголовку основного объекта раздела (рис. 9).

Рис. 9. — Выбор объекта привязываемых данных



В появившемся диалоговом окне с запросом о сохранении результатов нужно нажать на кнопку [Да].

После этого следует выбрать тип установки. Если демоданные не предполагается в дальнейшем изменять, достаточно их установить при первой установке пакета. Для этого нужно выбрать тип установки "Первичная установка". Привязываемые данные можно просмотреть, нажав на кнопку [Показать данные] (рис. 10). Также привязываемые данные можно отфильтровать, построив нужное условие фильтрации.

Рис. 10. — Привязанные данные демо-наполнения

The screenshot displays a software interface for managing web data. It is divided into two main sections: 'Свойства' (Properties) on the left and a data table on the right.

Свойства (Properties):

- Название: tsaWebData
- Объект: WebData
- Тип установки: Первичная установка
- Колонки: Includes buttons for 'Добавить', 'Изменить', and 'Удалить'.
- Table with columns: Название, Обя..., Ключ. The first row contains 'URL'.
- Buttons: 'Добавить новое условие' (Add new condition) with a checked checkbox.

Показывать данные (Show data): Всего записей: 2

Название	URL
bpmonline.com	https://www.similarweb.com/website/bpmonline.com
terrasoft.ru	https://www.similarweb.com/website/terrasoft.ru

At the bottom right, there are buttons for 'Сохранить' (Save) and 'Отмена' (Cancel).

Для окончательной привязки данных к пакету необходимо нажать на кнопку [Сохранить].